IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicants: Richard E. Anderson	**Examiner:** ARISTOCRATIS FOTAKIS
Serial No.: 09/052,673	**Art Unit:** 2611
Filed: March 31, 1998	**Docket:** END919980042US1 (11232)
For: DIGITAL AUDIO/VIDEO CLOCK RECOVERY ALGORITHM	**Dated:** December 21, 2011
Confirmation No.: 9401	

Commissioner for Patents
P. O. Box 1450
Alexandria, VA 22313-1450

APPEAL BRIEF

Sir:

Pursuant to 35 U.S.C. §134 and 37 C.F.R. §41.37, entry of this Appeal Brief in support of the Notice of Appeal filed October 21, 2011, in the above-identified matter is respectfully requested.

I. Statement of Real Party in Interest

The real party in interest in the above-identified patent application is the International Business Machines Corporation.

II. Statement of Related Appeals and Interferences

There are no other prior or pending appeals, interferences or judicial proceedings known to appellants, the appellants' legal representative, or assignee which may be related to, directly affect or be directly affected by or have a bearing on the Board's decision in this appeal.

III. Status of Claims

A. Claim Status

Claims 1-5, 7-10 and 13-15 are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the enablement requirement.

Claims 6, 11 and 12 are cancelled.

B. Appealed Claims

Claims 1-5, 7-10 and 13-15 are appealed. A clean copy of these claims is contained in Appendix A attached to this Appeal Brief.

IV. Status of Amendments

No Amendments have been filed after the Final Rejection of June 21, 2011.

V. Summary of Claimed Subject Matter

Concise explanation of the subject matter of Claim 1

Claim 1 is directed to a method of synchronizing the frequency of a local clock of a digital data decoder with the frequency of a program clock (Page 25, lines 11-17; Fig. 7). The decoder includes clock adjustment hardware (Page 25, lines 20 and 29; Fig. 2) for adjusting the local clock frequency and a processor having a software program (Page 25, line 29 – Page 26, Line 3; Fig. 8; Fig. 9) for adjusting the local clock frequency. The method of Claim 1 comprises the step of determining the difference between the local and program clock frequencies (Page 25, lines 19-26; Fig. 7), then adjusting the frequency at which the local clock oscillates so that said difference approaches zero (Page 25, line 28 – Page 26, line 19; Fig. 7). This adjusting includes the steps of: (i) using the clock adjustment hardware to adjust the local clock frequency until a threshold condition occurs (Page 25, line 28 – Page 26, line 19; Fig. 7; Page 14, lines 12-23; Page 26, lines 21-25; Page 26, line 26 – Page 28, line 14; Fig. 8; Fig. 9), and (ii) after the threshold condition occurs, using the software program of the processor to adjust the local clock frequency (Page 25, line 28 – Page 26, line 19; Fig. 7; Page 14, lines 12-23; Page 26, lines 21-25; Page 26, line 26 – Page 28, line 14; Fig. 8; Fig. 9).

Concise explanation of the subject matter of Claim 3

Claim 3 defines a method of synchronizing the frequency of a local clock of a digital data decoder with the frequency of a program clock (Page 25, lines 11-17; Fig. 7), wherein the local clock oscillates at a local clock frequency. The method of Claim 3 comprises the steps of: determining the difference between the local and program clock frequencies (Page 25. lines 19-26; Fig. 7), then adjusting the frequency at which the local clock oscillates so that said difference approaches zero (Page 25, line 28 – Page 26, line 3; Fig. 8; Fig .9); maintaining a local clock value based on the oscillations of the local clock (Page 26, lines 11-19; Fig. 7); receiving program clock data at the decoder which specify the program clock frequency (Page 25, lines 19-26; Page 26, lines 11-19; Fig. 7); maintaining a program clock value based on the program clock data received at the decoder (Page 25, lines 19-26; Page 26, lines 11-19; Fig. 7); determining if there is an absolute difference between the local clock value and the program clock value (Page 25, lines 19-26; Page 26, lines 11-19; Fig. 7).

The method of Claim 3 comprises the further step of, if there is an absolute difference between the local clock value and the program clock value, then adjusting the frequency at which the local clock oscillates so that said absolute difference approaches zero (Page 25, lines 19-26; Page 26, lines 11-19; Fig. 7). As described in Claim 3, the decoder includes clock adjustment hardware (Page 25, lines 20 and 29; Fig. 2) for adjusting the local clock frequency and a processor having a software program (Page 25, line 29 – Page 26, line 3; Fig. 8; Fig. 9) for adjusting the local clock frequency. Also, the step of adjusting the frequency of the local clock includes the steps of using the hardware to adjust the local clock frequency until a threshold condition occurs (Page 25, line 28 – Page 26, line 19; Fig. 7; Page 14, lines 12-23; Page 26, lines 21-25; Page 26, line 26 – Page 28, line 14; Fig. 8; Fig. 9), and after the threshold condition occurs, using the software program of the processor to adjust the local clock frequency (Page 25, line 28 – Page 26, line 19; Fig. 7; Page 14, lines 12-23; Page 26, lines 21-25; Page 26, line 26 – Page 28, line 14; Fig. 8; Fig. 9).

Concise explanation of the subject matter of Claim 7

Claim 7 is directed to a system for adjusting a local clock on a digital data decoder, wherein the clock oscillates at a local clock frequency. The system comprises a system time clock register (707) (Page 26, lines 11-19; Fig. 7) for maintaining a local clock value based on the oscillations of the local clock; means (Fig. 6; Page 17, lines 14-29; Page 24, lines 12-24; Page 25, lines 19-26) for receiving program clock data transmitted to the decoder that specify a program clock frequency; a program clock register (701) (Page 26, lines 11-19; Fig. 7) for maintaining a program clock value based on the clock data transmitted to the decoder; means (Page 25, lines 3-9) (Fig. 7) for determining if there is any difference between the local clock and the program clock frequencies; and a transport demultiplexer (Fig. 7) for determining if there is an absolute difference between the local clock value and the program clock value (707) (Page 26, lines 11-19; Fig. 7).

The system of Claim 7 further comprises means (800) (Fig. 8; Page 25, liNe 28 – Page 26, line 10) for adjusting the frequency at which the local clock oscillates, when there is a difference between the local clock and the program clock frequencies or an absolute difference between the local clock value and the program clock value, so that said difference approaches zero. As described in Claim 7, this means for adjusting the frequency at which the local clock oscillates includes hardware (Fig. 8; Fig. 3; Page 14, lines 21-23; Page 26, lines 3-10) for adjusting the local clock frequency until a threshold condition occurs, and a processor (Fig. 8; Page 25, line 29 – Page 26, line 3; Fig. 9) having a software program for adjusting the local clock frequency after the threshold condition occurs (Page 25, line 28 – Page 26, line 19; Fig. 7; Page 14, lines 12-23; Page 26, lines 21-25; Page 26, line 26 – Page 28, line 14; Fig. 8; Fig. 9).

Concise explanation of the subject matter of claim 13

Claim 13 defines a system for synchronizing the frequency of a local clock of a digital data decoder with the frequency of a program clock. This system comprises means (Page 25, lines 3-9) (Fig. 7) for determining if there is any difference between the local and program clock frequencies; and means (800) (Fig. 8; Page 25, line 28 – Page 26, line 10) for adjusting the frequency at which the local clock oscillates, when there is a difference between the local clock and the program clock frequencies, so that said difference approaches zero. The means for adjusting includes (i) hardware (Fig. 8; Fig. 3; Page 14, lines 21-23; Page 26, lines 3-10) on the decoder for adjusting the local clock frequency until a threshold condition occurs, and (ii) a processor (Fig. 8; Page 25, line 29 – Page 26, line 3; Fig. 9) on the decoder and having a software program for adjusting the local clock frequency after the threshold condition occurs (Page 25, line 28 – Page 26, line 19; Fig. 7; Page 14, lines 12-23; Page 26, lines 21-25; Page 26, line 26 – Page 28, line 14; Fig. 8; Fig. 9).

Identification of means plus function and step plus function under 35 U.S.C §112, and identification of structure, materials or acts described in the specification as corresponding to each claimed function

Claim 1

Claim 1 includes the following step plus function clauses.

determining the difference between the local and program clock frequencies,

Corresponding structure, material or acts

This is done by the demultiplexer shown in Fig. 6. (Page 25, lines 19-26; Fig. 7)

adjusting the frequency at which the local clock oscillates so that said difference approaches zero;

Corresponding structure, material or acts

This is done by the clock adjustment hardware shown in Fig. 8 and Fig. 3 and by the processor operating the software program shown in Figs. 8 and 9. (Page 25, line 28 – Page 26, line 19; Fig. 7)

using the clock adjustment hardware to adjust the local clock frequency until a threshold condition occurs,

Corresponding structure, material or acts

This is done by the clock adjustment hardware shown in Fig. 8 and Fig. 3. (Page 25, line 28 – Page 26, line 19; Fig. 7; Page 14, lines 12-23; Page 26, lines 21-25; Page 26, line 26 – Page 28, line 14; Fig. 8; Fig. 9)

after the threshold condition occurs, using the software program of the processor to adjust the local clock frequency.

Corresponding structure, material or acts

This is done by the processor operating the software program shown in Figs. 8 and 9. (Page 25, line 28 – Page 26, line 19; Fig. 7; Page 14, lines 12-23; Page 26, lines 21-25; Page 26, line 26 – Page 28, line 14; Fig. 8; Fig. 9)

Claim 3

Claim 3 includes the following step plus function clauses.

determining the difference between the local and program clock frequencies,

Corresponding structure, material or acts

This is done by the demultiplexer shown in Fig. 6. (Page 25, lines 19-26; Fig. 7)

adjusting the frequency at which the local clock oscillates so that said difference approaches zero;

Corresponding structure, material or acts

This is done by the clock adjustment hardware shown in Fig. 8 and Fig. 3 and by the processor operating the software program shown in Figs. 8 and 9. (Page 25, line 28 – Page 26, line 19; Fig. 7)

maintaining a local clock value based on the oscillations of the local clock;

Corresponding structure, material or acts

This is done by the demultiplexer shown in Fig. 6. (Page 26, lines 11-19; Fig. 7)

receiving program clock data at the decoder which specify the program clock frequency;

Corresponding structure, material or acts

This is done by the demultiplexer shown in Fig. 6. (Page 25, lines 19-26; Page 26, lines 11-19; Fig. 7)

maintaining a program clock value based on the program clock received at the decoder;

Corresponding structure, material or acts

This is done by the demultiplexer shown in Fig. 6. (Page 25, lines 19-26; Page 26, lines 11-19; Fig. 7)

determining if there is an absolute difference between the local clock value and the program clock value;

Corresponding structure, material or acts

This is done by the demultiplexer shown in Fig. 6. (Page 25, lines 19-26, Page 26, lines 11-19; Fig. 7)

if there is an absolute difference between the local clock value and the program clock value, then adjusting the frequency at which the local clock oscillates so that said absolute difference approaches zero;

Corresponding structure, material or acts

This is done by the clock adjustment hardware shown in Fig. 8 and Fig. 3 and by the processor operating the software program shown in Figs. 8 and 9. (Page 25, lines 19-26; Page 26, lines 11-19; Fig. 7)

using the hardware to adjust the local clock frequency until a threshold condition occurs;

Corresponding structure, material or acts

This is done by the clock adjustment hardware shown in Fig. 8 and Fig. 3. (Page 25, line 28 – Page 26, line 19; Fig. 7; Page 14, lines 12-23; Page 26, lines 21-25; Page 26, line 26 – Page 28, line 14; Fig. 8; Fig. 9)

after the threshold condition occurs, using the software program of the processor to adjust the local clock frequency.

Corresponding structure, material or acts

This is done by the processor operating the software program shown in Figs. 8 and 9. (Page 25, line 28 – Page 26, line 19; Fig. 7; Page 14, lines 12-23; Page 26, lines 21-25; Page 26, line 26 – Page 28, line 14; Fig. 8; Fig. 9)

Claim 7

Claim 7 includes the following means plus function clauses.

means for receiving program clock data transmitted to the decoder that specify a program clock frequency;

Corresponding structure, material or acts

This is done by the demultiplexer shown in Fig. 6. (Fig. 6; Page 17, lines 14-29; Page 24, lines 12-24; Page 25, lines 19-26)

means for determining if there is any difference between the local clock and the program clock frequencies;

Corresponding structure, material or acts

This is done by the demultiplexer shown in Fig. 6. ((Page 25, lines 3-9) (Fig. 7)

means for adjusting the frequency at which the local clock oscillates, when there is a difference between the local clock and the program clock frequencies or an absolute difference between the local clock value and the program clock value, so that said difference approaches zero;

Corresponding structure, material or acts

This is done by the clock adjustment hardware shown in Fig. 8 and Fig. 3 and by the processor operating the software program shown in Figs. 8 and 9. (800) (Fig. 8; Page 25, line 28 – Page 26, line 10)

Claim 13

Claim 13 includes the following means plus function clauses.

means for determining if there is any difference between the local and program clock frequencies;

Corresponding structure, material or acts

This is done by the demultiplexer shown in Fig. 6. ((Page 25, lines 3-9) (Fig. 7)

means for adjusting the frequency at which the local clock oscillates, when there is a difference between the local clock and the program clock frequencies, so that said difference approaches zero,

Corresponding structure, material or acts

This is done by the clock adjustment hardware shown in Fig. 8 and Fig. 3 and by the processor operating the software program shown in Figs. 8 and 9. (800) (Fig. 8; Page 25, line 28 – Page 26, line 10)

VI. Grounds of Rejection to be Reviewed On Appeal

Appellant asks that the following grounds of rejection be reviewed:

Whether Claims 1-5, 7-10 and 13-15 fail to comply with the enablement requirement of 35 U.S.C. 112, first paragraph.

VII. Argument

A. Introduction

Claims 1-5, 7-10 and 13-15, which are all of the pending claims, are rejected under 35 U.S.C. 112, first paragraph, on the grounds that the specification is not sufficiently enabling. None of the claims is rejected over the prior art.

The rejection of the Claims should be reversed because the specification fully enables those of ordinary skill in the art to practice the claimed invention.

In support of the rejection, the Examiner argued, with regard to the independent Claims 1, 3, 7 and 13, that "[T]he specification… never teaches determining the difference between the local and program clock frequencies before the step of adjusting [the frequency of the local clock]." In particular, the Examiner argues that one of ordinary skill in the art would not be able to determine this difference by using the disclosed PCR and STC values (Final Rejection dated June 21, 2011, Page 2, lines 13-20).

Claim 1 is representative of Claims 1, 3, 7 and 13.

Appellants respectfully disagree with the Examiner for a number of reasons. One reason is because the specification expressly does teach how to determine the difference between the program and system clock frequencies by using the PCR and STC values. In addition, procedures for determining the difference between the program and system clock frequencies are known in the art.

B. Discussion

1. The invention

Appellants' invention, generally, relates to synchronizing the frequency of a local clock of a digital data decoder with the frequency of a program clock. As discussed in detail in the present application, in situations where compressed data are transmitted to a decoder, the frequency of the program clock and the frequency of the local clock on the decoder need to be kept reasonably close together in order to properly decompress the data.

The present invention addresses this issue, and does so in a manner that places reduced demand on the processor used to keep the clock frequencies together. In accordance with Appellants' invention, under some circumstances, the frequencies of the two clocks are brought together using hardware, and in other circumstances, software, running on the processor, is used to bring these two frequencies closer together. More specifically, the hardware is used to bring the two frequencies together before a threshold condition occurs; and after this threshold condition occurs, software is used to bring the two frequencies together. Figure 7 of the application illustrates one embodiment of the hardware that may be used, and Figure 9 of the application illustrates a software procedure that may be used

With this procedure of Applicants' invention, the processor is only called upon under specific conditions to address the frequency differences between the program and local clocks.

Applicants' invention uses information contained in the Program Clock Reference (PCR) fields transmitted to the decoder, and also information generated about the local clock, referred to as the System Time Clock (STC) value. The PCR fields are well known in the field and are discussed in the Background section of the application.

2. The rejection

As mentioned above, the Examiner argued, with regard to the independent Claims 1, 3, 7 and 13, that "[T]he specification… never teaches determining the difference between the local and program clock frequencies before the step of adjusting [the frequency of the local clock]." In particular, the Examiner argues that one of ordinary skill in the art would not be able to determine this difference by using the disclosed PCR and STC values (Final Rejection dated

June 21, 2011, Page 2, lines 13-20). Also, with regard to Claims 2 and 3, the Examiner requested that Applicants indicate which portion of the specification has support on the claimed limitation of "maintaining a program clock value based on the program clock signals received at the decoder" (Final Rejection dated June 21, 2011, Page 3, lines 6-8) (emphasis in the original).

3. Response to the Examiner's arguments.

With respect to the Examiner's specification objection to the independent Claims 1, 3, 7 and 13, Claim 1 is representative of Claims 3, 7 and 13.

As indicated above, the specification expressly does teach how to determine the differences between the program and system clock frequencies by using the PCR and STC values. One factor that the Examiner appears not appear to fully appreciate is that the PCR value can be considered as the reference clock frequency. This is explained in the specification on Page 24, lines 16-22, which state:

> "Program Clock Reference, or PCR, is a 42 bit field. It is coded in two parts, a PCR Base having a 22-bit value in units of 90 kHz, and…a PCR extension having a 9-bit extension in units of 27 MHz. 27 MHz is the system lock frequency."

Thus, the PCR can be considered as the frequency of the Program Clock, expressed in defined units.

Then, the application expressly states how the difference between the frequencies of the program and system clocks can be determined. First, the application explains how this is done in the prior art. On page 25, lines 3-9, the specification states that: "…the first PCR initialized the counter in a clock generation, and subsequent PCR values are compared to clock values for fine adjustment. The differen[ce] between the PCR and the local clock can be used to drive a voltage controlled oscillator…to speed up or slow down the local clock."

The specification then explains how the difference between the frequencies of the program and system clocks is determined in an embodiment of the invention. On page 25, lines 21-26, the specification states that: "The transport demultiplexor of the invention extracts Program clock References (PCRs) from the indicated PID, calculates the offset from the current System Time Clock (STC) value, and compares it against a threshold defined by the application to determine if clock frequency correction is determined.

From these disclosures, it is readily apparent that the PCR can be considered as the system clock frequency (expressed in some defined unit such as 27MgH), that the SCT value is the local clock frequency (expressed in some unit), and that the difference between the frequencies of these two clocks is determined by "calculate[ing] the offset" between these two values. These frequency values are not complicated numbers, and it is not difficult to determine the "offset" or difference between the two numbers. Appellants submit that the necessary calculating is well within those of ordinary skill in the art.

In addition to the foregoing, procedures for determining the difference between the program and system clock frequencies are well known in the art. The present application does not have to teach those of ordinary skill in the art how to do this. The present invention is described in the specification as an improvement on prior art procedures that determine that difference and then attempt to adjust the frequency of the system clock to be synchronous with the program clock.

As is clear from a reading of the application, Appellants are not claiming that they have developed any new way <u>to determine</u> the frequency difference between the program and system clocks. Instead, what Appellants have done is to provide a new procedure <u>for reducing</u> that difference . This procedure for reducing the frequency difference is explained and taught in detail in the present application.

It thus follows that the Examiner's rejection of the claims based on the limitation "determining the difference between the local and program clock frequencies, then adjusting the frequency at which the local clock oscillates so that the difference approaches zero," is not proper.

In response to the Examiner request that Applicants indicate which portion of the specification has support on the claimed limitation of "maintaining a program clock value based on the program clock signals received at the decoder," Appellants note that this aspect of the disclosed invention is discussed at several places in the specification. As noted previously, the specification, on page 25, lines 21-26, states that: "The transport demultiplexor of the invention extracts Program clock References (PCRs) from the indicated PID…" Further, as explained on Page 26, lines 13-16, a Program Clock Recovery (PCR) 701 is provided. It is very clear that the PCRs extracted by the demultiplexor are stored in this PCR register. A reference to the "stored" PCR values is also given on Page 27, line 9.

Given these express disclosures in the specification, those of ordinary skill in the art would have no difficulty in practicing the claimed limitation of "maintaining a program clock values based on the program clock signals received at the decoder." Accordingly, the Examiner's rejection of Claims 2 and 3 based on this limitation is also not proper.

Appellants respectfully submit that it is critical to guard against requiring that the specification discloser more than what the law requires. While 35 U.S.C. 112, does require that the specification enable the claimed invention, this enablement requirement is only that the specification enable those of ordinary skill in the art. As the Court of Appeals for the Federal Circuit explained in Spectra-Physics v. Coherent, 3 USPQ2d 1737 (Fed. Cir. 1987):

> "The essence of [the enablement requirement] is that a specification shall disclose an invention in such a manner as will enable one skilled in the art to make and utilize it." Spectra-Physics v. Coherent, supra at 1742.

In the present case, those of ordinary skill in the art are able to make and utilize the invention.

VII. Conclusion

For the reasons advanced above, the claim limitations to which the Examiner has specifically objected are, in fact, fully enabled by the specification. Specifically, the claim limitation "determining the difference between the local and program clock frequencies, then adjusting the frequency at which the local clock oscillates so that the difference approaches zero," which occurs in the independent Claims 1, 3, 7 and 13 is fully enabled. Also, the claim limitation "maintaining a program clock value based on the program clock signals received at the decoder," which is present in Claims 2 and 3, is also properly disclosed and enabled by the specification. It follows that the rejection of the Claims1-5, 7-10, and 13-15 under 35 U.S.C. 112, first paragraph, is not proper, and the Board of Appeals is respectfully asked to reverse this rejection.

VIII. Claims Appendix

A clean copy of Claims 1, 2, 4-8, 10-14 and 16-20 are contained in Appendix A to this Appeal Brief.

IX. Evidence Appendix

Appellants are not relying on any affidavits, extrinsic documents or extrinsic evidence.

X. Related Proceedings Appendix

As indicated above, there are no other prior or pending appeals, interferences or judicial proceedings known to appellants, the appellants' legal representative, or assignee which may be related to, directly affect or be directly affected by or have a bearing on the Board's decision in this appeal.

Respectfully submitted,

/John S. Sensny/

Dated: December 21, 2011

John S. Sensny
Registration No. 28,757
Attorney for Applicants

Enclosure: Appendix A

APPENDIX A

1. A method of synchronizing the frequency of a local clock of a digital data decoder with the frequency of a program clock, wherein the decoder includes clock adjustment hardware for adjusting the local clock frequency and a processor having a software program for adjusting the local clock frequency, the method comprising the steps of:

determining the difference between the local and program clock frequencies, then adjusting the frequency at which the local clock oscillates so that said difference approaches zero; including the steps of:

i) using the clock adjustment hardware to adjust the local clock frequency until a threshold condition occurs, and

after the threshold condition occurs, using the software program of the processor to adjust the local clock frequency.

2. A method according to Claim 1, wherein the local clock oscillates at the local clock frequency, the method further comprising the steps of:

maintaining a local clock value based on the oscillations of the local clock;

receiving program clock data at the decoder which specify the frequency of the program clock;

maintaining a program clock value based on the program clock data received at the decoder;

determining if there is an absolute difference between the local clock value and the program clock value;

if there is an absolute difference between the local clock value and the program clock value, then adjusting the frequency at which the local clock oscillates so that said absolute difference approaches zero.

3. A method of synchronizing the frequency of a local clock of a digital data decoder with the frequency of a program clock, wherein the local clock oscillates at a local clock frequency, the method comprising the steps of:

determining the difference between the local and program clock frequencies, then adjusting the frequency at which the local clock oscillates so that said difference approaches zero;

maintaining a local clock value based on the oscillations of the local clock;

receiving program clock data at the decoder which specify the program clock frequency;

maintaining a program clock value based on the program clock data received at the decoder;

determining if there is an absolute difference between the local clock value and the program clock value;

if there is an absolute difference between the local clock value and the program clock value, then adjusting the frequency at which the local clock oscillates so that said absolute difference approaches zero;

wherein the decoder includes clock adjustment hardware for adjusting the local clock frequency and a processor having a software program for adjusting the local clock frequency, and wherein the step of adjusting the frequency of the local clock includes the steps of:

using the hardware to adjust the local clock frequency until a threshold condition occurs; and

after the threshold condition occurs, using the software program of the processor to adjust the local clock frequency.

4. A method according to Claim 3, wherein the threshold condition is a function of the difference between the local clock value and the program clock value.

5. A method according to Claim 3, wherein the step of using the software program of the processor to adjust the local clock frequency includes the steps of:

monitoring for the occurrence of the threshold condition; and

transmitting a signal to the processor when the threshold condition occurs.

7. A system for adjusting a local clock on a digital data decoder, wherein the clock oscillates at a local clock frequency, the system comprising:

a system time clock register for maintaining a local clock value based on the oscillations of the local clock;

means for receiving program clock data transmitted to the decoder that specify a program clock frequency;

a program clock register for maintaining a program clock value based on the clock data transmitted to the decoder;

means for determining if there is any difference between the local clock and the program clock frequencies;

a transport demultiplexer for determining if there is an absolute difference between the local clock value and the program clock value; and

means for adjusting the frequency at which the local clock oscillates, when there is a difference between the local clock and the program clock frequencies or an absolute difference between the local clock value and the program clock value, so that said difference approaches zero;

wherein the means for adjusting the frequency at which the local clock oscillates includes:

hardware for adjusting the local clock frequency until a threshold condition occurs; and

a processor having a software program for adjusting the local clock frequency after the threshold condition occurs.

8. A system according to Claim 7, wherein the threshold condition is a function of the difference between the local clock value and the program clock value.

9. A system according to Claim 7, wherein the processor is not used to adjust the local clock frequency until the threshold condition occurs.

10. A system according to Claim 7, said hardware includes:

a threshold register for monitoring for the occurrence of the threshold condition; and

means for transmitting a signal to the processor when the threshold condition occurs.

13. A system for synchronizing the frequency of a local clock of a digital data decoder with the frequency of a program clock, comprising:

means for determining if there is any difference between the local and program clock frequencies; and

means for adjusting the frequency at which the local clock oscillates, when there is a difference between the local clock and the program clock frequencies, so that said difference approaches zero, wherein the means for adjusting includes

i) hardware on the decoder for adjusting the local clock frequency until a threshold condition occurs, and

ii) a processor on the decoder and having a software program for adjusting the local clock frequency after the threshold condition occurs.

14. A system according to Claim 13, wherein the local clock has a local clock value and the program clock has a program clock value, and the threshold condition is a function of the difference between the local clock value and the program clock value.

15. A system according to Claim 13, wherein said hardware includes

a threshold register for monitoring for the occurrences of the threshold condition; and

means for transmitting a signal to the processor when the threshold condition occurs.